

Serial No.: 10/749,271

Amendment dated: January 8, 2007

Reply to Office Action dated: September 6, 2006

JAN 08 2007

**REMARKS/ARGUMENTS**

Claims 1-30 are pending in this application. Claims 1-2, 4, and 7 are rejected under 35 U.S.C. §102(b) as being anticipated by Merchant et al., U.S. Patent No. 6,385,715. Claims 3, 9-15, 18-23, 25-27, and 29-30 are rejected under U.S.C. §103(a) as being unpatentable over Merchant et al., U.S. Patent No. 6,385,715. Claims 5-6 and 16-17 are rejected under U.S.C. §103(a) as being unpatentable over Merchant et al., U.S. Patent No. 6,385,715, in view of Merchant et al., U.S. Patent No. 6,163,838 (hereinafter referred to as '838). Claims 8, 24, and 28 are rejected under U.S.C. §103(a) as being unpatentable over Merchant et al., U.S. Patent No. 6,385,715, in view of Topham et al, U.S. Patent No. 6,944,853.

Applicants respectfully submit the cited references do not teach, suggest or describe at least "An adaptive replay system comprising:... *a selector device coupled to said staging area to place said instruction in an optimal position within said replay loop...*" (e.g., as described in claim 1).

The Office Action asserts Merchant teaches such a selector device at column 6, lines 7-25, further including citations to element 150 and 154. *See* Office Action dated 4/20/2006, paragraph 10, page 4. Applicants disagree.

Element 150 of Merchant is described as a checker, while element 154 is described as a replay queue *unloading* controller (*emphasis added*). The cited section states:

If the checker 150 determines that the instruction has not executed properly, the instruction will then be returned to multiplexer 116 to be replayed (*i.e.*, re-executed). Each instruction to be replayed will be returned to mux 116 via one of two paths. Specifically, if the checker 150 determines that the instruction should be replayed, the Replay Queue Loading Controller 154 determines whether the instruction should be sent through a replay loop 156 including staging queues E and F, or whether the instruction should be temporarily stored in a replay queue 170 before returning to mux 116.

Serial No.: 10/749,271

Amendment dated: January 8, 2007

Reply to Office Action dated: September 6, 2006

Instructions routed via the replay loop 156 are coupled to mux 116 via line 161. Instructions can also be routed by controller 154 for temporary storage in replay queue 170 (prior to replay). The instructions stored in replay queue 170 are output or unloaded under control of replay queue unloading controller 179. The instructions output from replay queue 170 are coupled to mux 116 via line 171. The operation of replay queue 170, Replay Queue Loading Controller 154 and Replay Queue Unloading Controller 179 are described in detail below.

Applicants submit the cited section does not teach the relevant limitations; indeed, the cited section does not refer to optimally rearranging the order of instructions at all. The first sentence of the cited section introduces the “re-execut[ion]” process. Specifically, if the checker 150 determines the instruction did not execute properly, it is returned to the multiplexer 116. To do this, *a conditional determination* is made which decides to send the instruction down one of two paths. The first outcome of this *conditional determination* sends the instruction to the replay loop 156, while the second outcome sends it temporarily to the replay queue 170 before sending it to the mux 116. Instructions stored in replay queue are sent to controller 179, and are coupled to mux 116 (described above) by line 171.

Therefore, Applicants submit the heart of the cited section is this two-outcome *conditional determination* of where to send an instruction that hasn’t executed properly. However, making a conditional determination upon which one of two steps may be followed is not the same as placing an instruction in an optimal position *within a replay queue* (as described in embodiments of the present application) at all. Indeed, the cited section does not discuss *placement* of an instruction for any reason anywhere (other than re-sending the instruction as discussed above). In order to support a proper rejection, the cited section must describe at least placing an instruction in an optimal position with a replay loop. The Merchant reference does not. Applicants submit for at least this reason, this section is inadequate to support a proper

Serial No.: 10/749,271

Amendment dated: January 8, 2007

Reply to Office Action dated: September 6, 2006

rejection of independent claim 1.

The Office Action further argues element 150 determines if a replay is necessary and element 154 determines which replay path will be used for the instruction. Therefore, it asserts, element 154 places the instruction in an optimal position for the processor by either sending it through the staging queues E and F, or sending it to the replay queue where it could be delayed. *See Office Action 9/6/2006, paragraph 42.* Applicants maintain this assertion fails to support a proper rejection for reasons similar to those discussed above.

Specifically, placing an instruction in an optimal position *for the processor* is not the same as placing an instruction in an optimal position *within said replay loop* (e.g., as described in embodiment of the present application). As discussed above, the cited reference makes a conditional determination entailing either returning an instruction to the replay loop 156 to be repeated or sending it temporarily to the replay queue 170 before sending it to the mux 116. Such a determination or operation entails either repeating the instruction as a whole, or sending it temporarily to a replay queue 170. It does not entail placing an instruction in an optimal position *within said replay loop* (e.g., as described in embodiment of the present application). Applicants maintain in order to support a proper rejection, the cited section must describe at least placing an instruction in an optimal position *within a replay loop*.

Merchant '838 fails to make up for the deficiencies of Merchant '715. Merchant '838 is also directed toward a replay system for the purpose of replaying instructions, and similar to above fails to describe at least a selector device's ability to *place an instruction in an optimal position within a replay loop* anywhere as described in embodiments of the present application.

Finally, Topham fails to make up for the deficiencies of Merchant '715 as well. Topham

Serial No.: 10/749,271

Amendment dated: January 8, 2007

Reply to Office Action dated: September 6, 2006

is directed toward the predicated execution of instructions in processors, but does not describe at least a selector device's ability to *place an instruction in an optimal position within a replay loop* anywhere as described in embodiments of the present application.

Therefore, since for at least the preceding reasons each and every limitation is not taught or suggested in the cited references, Applicants submit they are inadequate to support proper 35 U.S.C. §102(b) and §103(a) rejections, and independent claim 1 should be allowed. Independent claims 13, 23, and 27 contain similar allowable limitations. Claims 2-12, 14-22, 24-26, and 28-30 depend from allowable independent claims and therefore are allowable as well.

For at least all the above reasons, the Applicants respectfully submit that this application is in condition for allowance. A Notice of Allowance is earnestly solicited.

The Examiner is invited to contact the undersigned at (408) 975-7500 to discuss any matter concerning this application.

The Office is hereby authorized to charge any additional fees or credit any overpayments under 37 C.F.R. §1.16 or §1.17 to Deposit Account No. 11-0600.

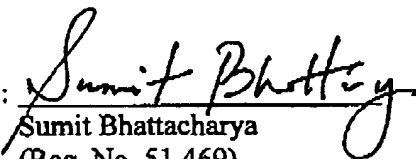
Respectfully submitted,

KENYON & KENYON LLP

Dated: January 8, 2007

KENYON & KENYON LLP  
333 W. San Carlos Street, Suite 600  
San Jose, CA 95110

Telephone: (408) 975-7500  
Facsimile: (408) 975-7501

By:   
Sumit Bhattacharya  
(Reg. No. 51,469)  
Attorneys for Intel Corporation